

Artificial Intelligence Internship Project for Python Skills Banking System Project

Sina YadeMellat / Deadline: 13 Ordibehesht 1403



صورت مسئله:

طراحی سیستم بانکی که شامل ادمین و مشتریان و ارتباط آن ها با هم می باشد.

کاربر Admin:

این کاربر باید با رمز عبور مجاز وارد سیستم شود (به طور پیش فرض میتواند یک فایل ساده شامل یک عدد ۸ رقمی برای ادمین در نظر بگیرید که هنگام ورود رمز را از فایل خوانده و صحت آن را بررسی کنید). ویژگی های

ادمین شامل موارد زیر است:

- first_name
- family_name
- national_code
- home_town
- password

ادمین دسترسی های زیر را دارد:

- تغییر رمز عبور
- ایجاد بانک
- ایجاد شعبه های بانک ها
- مشخص کردن بودجه هر شعبه
- ایجاد حساب بعد از درخواست افتتاح حساب از سمت مشتری
- دسترسی به اطلاعات مشتریان

در کل سیستم، فقط یک ادمین وجود دارد.



کاربر Customer:

مشتری ها رمز ورود ندارند و صرفا با کدملی و یا با شماره حساب خود می توانند وارد سیستم شده و method هایی که در ادامه ذکر شده است را میتوانند داشته باشد.

ویژگی های مشتری شامل موارد زیر است:

- first_name
- family_name
- national_code

- home_town
- account_number: list
- loan_number: list

مشتری میتواند:

1. درخواست افتتاح حساب کند.
2. اطلاعات حساب/حسابهای خود را ببیند.
3. برداشت از حساب کند.
4. واریز به حساب کند.
5. درخواست وام دهد.

a. درخواست وام برای هر مشتری از یک شعبه فقط و فقط یک بار باید صورت بگیرد. یعنی اگر آن مشتری چندین حساب در یک شعبه از بانک را داشته باشد نمیتواند به ازای هر حساب وام بگیرد(فقط یک بار وام باید برای آن لحاظ شود).

b. وام درخواستی باید در حد بودجه ی آن شعبه نیز باشد. هر مقداری برای وام مجاز نیست.

علاوه بر این دو نوع کاربر، موجودیت های دیگری هم در این سیستم برای ارتباط بهتر وجود دارد که براساس طراحی شما ممکن است تعداد آن ها کم یا زیاد شود اما به صورت کلی:

Bank:

name, bank_id

Branch:

name, branch_id, number_of_customers, budget, city_name

Account:

account_number(8 digits, unique), account_amount, account_owner

(optional)

Loan:

loan_number, loan_amount, customer_id, account_id, branch_id

نکته هایی که حتما باید رعایت کنید:

- ★ از گیت استفاده کنید و پروژه ی نهایی را بر روی github قرار دهید و لینک repository را برای من ارسال کنید.
- ★ برنامه باید به صورت modular نوشته شود.
- ★ در صورت لزوم حتما از ارث بری استفاده شود.
- ★ نکات clean code در کل پروژه و اصول SOLID در نوشتن class ها رعایت شود.
- ★ خطاهای احتمالی را با exception کنترل کنید.
- ★ در این پروژه نیازی به استفاده از database نمی باشد اما در صورتی که قادر به پیاده سازی آن هستید به عنوان bonus در نظر گرفته میشود.
- ★ در صورت امکان تست نویسی شود. (bonus)

If any question, please feel free to ask, Telegram: @Samin_Yadollahi

Good Luck ^_^